Phân Tích Dữ Liệu Thực Tế với Python Bài 4.1: Làm Quen Matplotlib



Quang-Khai Tran, Ph.D CyberLab, 02/2023



(Ånh: Internet)





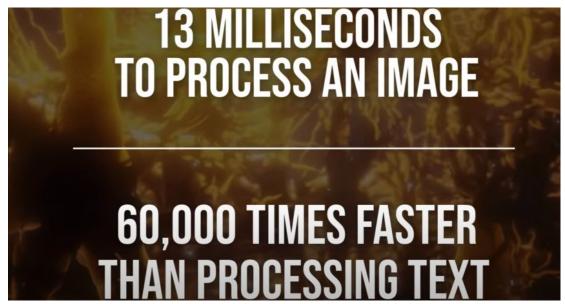


## Nội dung



- 1. Giới thiệu thư viện Matplotlib
- 2. Sử dụng hàm plot()
- 3. Các thao tác cơ bản khác
- 4. Demo & Bài tập

"A Picture Is Worth A Thousand Words" (since 1910s, see Wikipedia)



Source: https://www.youtube.com/watch?v=GpP0EbSMRpg&ab channel=LeilaGharani



- 1.1. Giới thiệu
- 1.2. Các loại biểu đồ thông dụng
- 1.3. Cấu trúc một biểu đồ

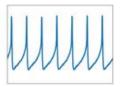


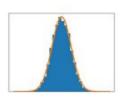
Matplotlib là thư viện phổ biến nhất để trực quan hóa dữ liệu trên Python

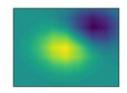


#### Matplotlib: Visualization with Python

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python.





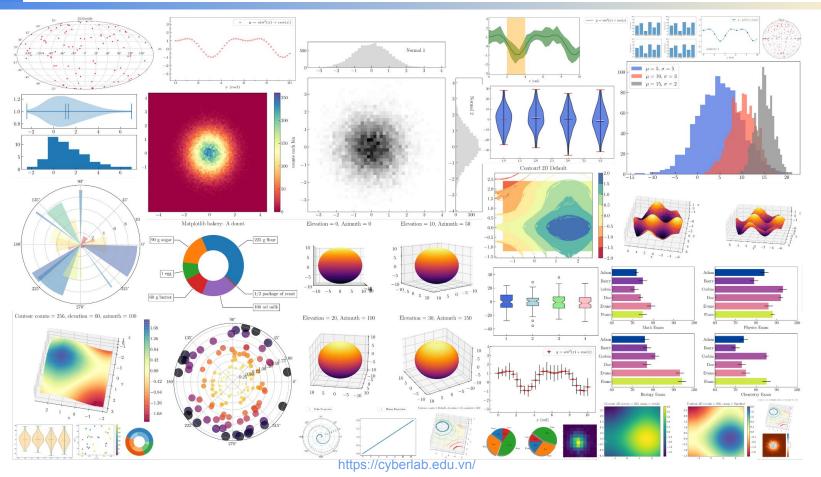




Matplotlib makes easy things easy and hard things possible.

https://matplotlib.org/stable/index.html







#### Các trang web tham khảo hữu ích:

- Trang chủ: <a href="https://matplotlib.org/">https://matplotlib.org/</a>
- Trang chủ phiên bản stable: <a href="https://matplotlib.org/stable/index.html">https://matplotlib.org/stable/index.html</a>
- Công đồng: <a href="https://discourse.matplotlib.org/c/community/11">https://discourse.matplotlib.org/c/community/11</a>
- Matplotlib Gallery: <a href="https://matplotlib.org/stable/gallery/index.html">https://matplotlib.org/stable/gallery/index.html</a>
- Cheatsheet: <a href="https://github.com/matplotlib/cheatsheets#cheatsheets">https://github.com/matplotlib/cheatsheets#cheatsheets</a>
- 50 loại plots cho Machine Learning: <a href="https://www.machinelearningplus.com/plots/top-50-matplotlib-visualizations-the-">https://www.machinelearningplus.com/plots/top-50-matplotlib-visualizations-the-</a>
  <a href="mailto:-master-plots-python/">-master-plots-python/</a>

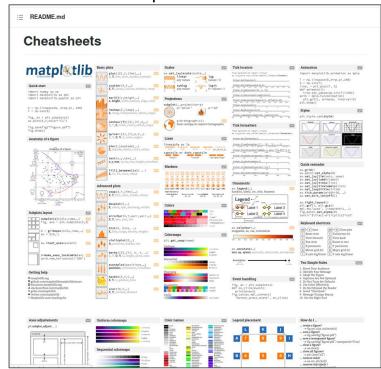


#### Matplotlib Gallery



Link: https://matplotlib.org/stable/gallery/index.html

#### Matplotlib Cheatsheets



Link: <a href="https://github.com/matplotlib/cheatsheets#cheatsheets">https://github.com/matplotlib/cheatsheets#cheatsheets</a>



#### Để sử dụng matplotlib trong Python:

cần import thư viện matplotlib và module pyplot.

```
import matplotlib
import matplotlib.pyplot as plt

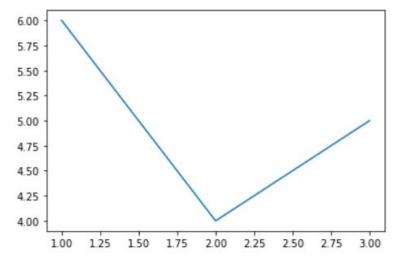
print(matplotlib.__version__)
print(matplotlib)
print(plt)
```



#### Ví dụ 1:

```
# Import và cách sử dụng đơn giản
from matplotlib import pyplot as plt

x = [1,2,3]
y = [6,4,5]
plt.plot(x,y)
plt.show()
```

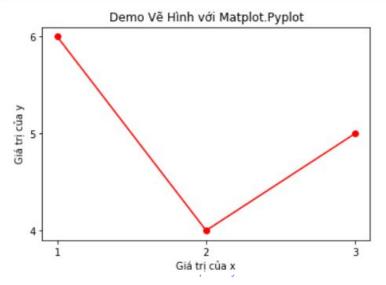




#### Ví du 2:

```
# Import và cách sử dụng đơn giản
from matplotlib import pyplot as plt

plt.plot([1,2,3], [6,4,5], color="red", marker="o")
plt.title("Demo Vẽ Hình với Matplot.Pyplot")
plt.xlabel("Giá trị của x")
plt.ylabel("Giá trị của y")
plt.xticks(x)
plt.yticks(y)
plt.show()
```

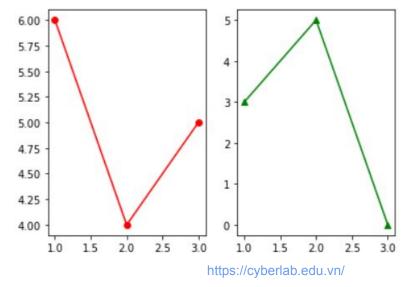




#### Ví dụ 3:

```
# Subplots
from matplotlib import pyplot as plt
plt.subplot(1,2,1)
plt.plot([1,2,3], [6,4,5], color="red", marker="o")

plt.subplot(1,2,2)
plt.plot([1,2,3], [3,5,0], color="green", marker="^")
plt.show()
```



## 1.2 Các loại biểu đồ thông dụng



#### Các biểu đồ cơ bản:

#### Sẽ làm quen ở buổi tiếp theo

Bar	Histogram	Scatter	Area	Pie
plt.bar()	plt.hist()	plt.scatter()	plt.stackplot()	plt.pie()











## 1.2 Các loại biểu đồ thông dụng



#### Các biểu đồ nâng cao:

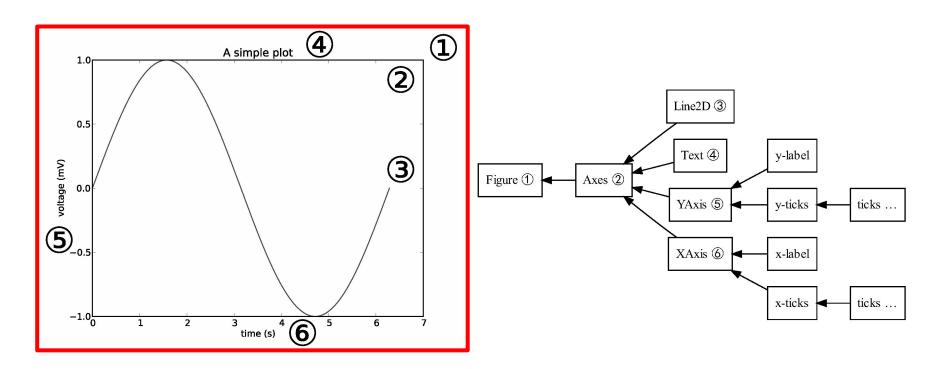
- Biểu đồ bubble
- ❖ Biểu đồ box
- ❖ Biểu đồ donut/nested-donut
- Biểu đồ radar
- Biểu đồ nhiệt
- ❖ Biểu đồ bản đồ

. . .

Sẽ làm quen ở Tuần 07/08



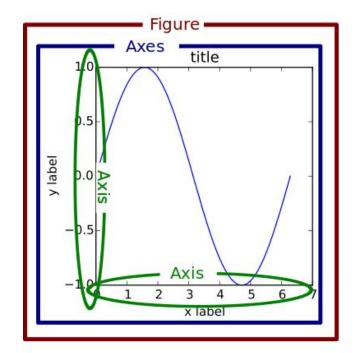
Biểu đồ của Matplotlib có cấu trúc phân tầng





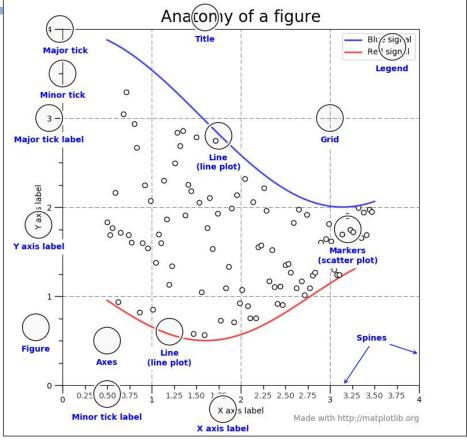
#### Cấu trúc phân tầng trong matplotlib

- Một đối tượng "biểu đồ" gọi là một Figure
- Các biểu đồ con (subplot) gọi là Axes



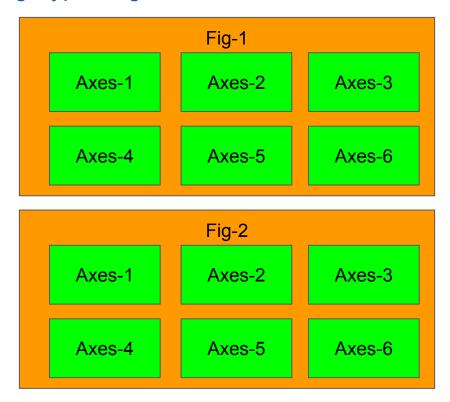


## Tên của một số đối tượng chính trong biểu đồ





#### Phân chia "lưới" trong Pyplot.Figure





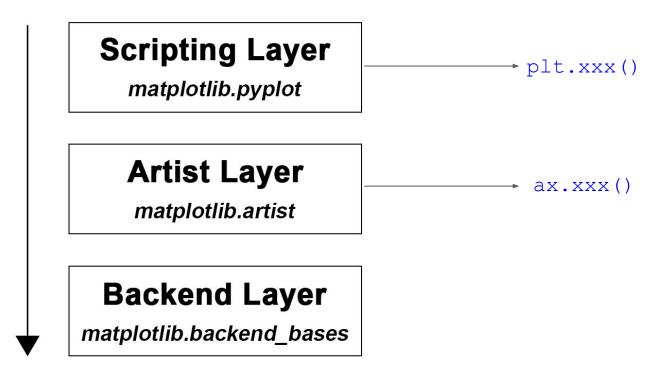
#### Đặc biệt:

- Có thể tạo nhiều Figure trong một hình biểu diễn biểu đồ
- Có thể dùng phép lặp để truy cập các Axes

```
# Dùng phép lặp để truy cập các Axes
fig,axes = plt.subplots(3, 2, constrained_layout=False)
print(axes.shape)
for i in range(3):
    for j in range(2):
        ax = axes[i][j]
        ax.plot([1,2,3], [6,4,5])
plt.show()
(3, 2)
```



03 tầng của thư viện Matplotlib





#### Phân biệt cách dùng Pyplot và Figure:

- ❖ Tầng Artist
  - Figure/Axes là dạng object-oriented API (hướng đối tượng) phù hợp hơn để làm việc trong Python và DS
  - Figure/Axes phù hợp hơn để vẽ các biểu đồ phức hợp, phức tạp

#### ❖ Tầng Scripting:

- Pyplot được coi là một API vẽ biểu đồ giống kiểu Matlab-style (chuyên dùng để làm việc với các phép toán trong nhiều lĩnh vực)
- Đối tượng Pyplot (plt) chính là đối tượng "hiện tại" trong phân tầng,
   (luôn trỏ tới đối tượng Axes hiện tại trong Figure hiện tại, hay subplot hiện tại)
- Pyplot có thể coi là một cách "lười" trong việc truy cập Figure/Axes



# Phần 2. Sử dụng hàm plot()

- 2.1. Tổng quan về hàm plot()
- 2.2. Sử dụng color/marker/line-style
- 2.3. Gán nhãn cho thông số trong biểu đồ

## 2.1 Tổng quan về hàm plot()



Tham khảo API:

https://matplotlib.org/stable/api/ as gen/matplotlib.pyplot.plot.html

```
plt.plot(x, y, color='green', marker='o', linestyle='dashed',
... linewidth=2, markersize=12)
```

```
plt.plot(x1, y1, 'g^', x2, y2, 'g-')
```

## 2.1 Tổng quan về hàm plot()



## Các tham số quan trọng:

Tham số	Mô tả
color	Màu
marker, markersize	Kiểu và kích thước của marker
linestyle or ls	Kiểu đường
linewidth or lw	Kích thước của đường
alpha	Độ mờ - rõ của màu tô (0:1)
label	Nhãn của chuỗi dữ liệu được vẽ

## 2.2 Sử dụng color/mark<u>er/line-stvle</u>



#### Sử dụng color

- https://matplotlib.org/stable/ga llery/color/named\_colors.html
- https://matplotlib.org/2.0.2/api/ colors\_api.html

character	color	
'b'	blue	
'g'	green	
'r'	red	
'c'	cyan	
'm'	magenta	
'y'	yellow	
'k'	black	
'W'	white	





#### Sử dụng marker

https://matplotlib.org/stable/ap i/markers\_api.html?highlight= markers#module-matplotlib.m arkers

#### matplotlib.markers

Functions to handle markers; used by the marker functionality of plot, scatter, and errorbar.

All possible markers are defined here:

http

marker	symbol	description
н . н	•	point
","	0	pixel
"0"	•	circle
"V"	▼	triangle_down
п^п	<b>A</b>	triangle_up
"<"	◀	triangle_left
">"	•	triangle_right
"1"	Y	tri_down
"2"	1	tri_up
"3"	~	tri_left
"4"	<b>&gt;</b>	tri_right
"8"	•	octagon
"S"		square
"p"	•	pentagon
"P"	+	plus (filled)
u*n	*	star
"h"	•	hexagon1
"H"	•	hexagon2
"+"	+	plus
"X"	×	X
"X"	*	x (filled)
"D"	•	diamond
"d"	•	thin_diamond



#### Sử dụng marker

https://matplotlib.org/ i/markers\_api.html? markers#module-m arkers

"D"	•	diamond
"d"	•	thin_diamond
"   "	T)	vline
п_п	-	hline
0 (TICKLEFT)	-	tickleft
1 (TICKRIGHT)	_	tickright
2 (TICKUP)	I	tickup
3 (TICKDOWN)	î.	tickdown
4 (CARETLEFT)	4	caretleft
5 (CARETRIGHT)	<b>&gt;</b>	caretright
6 (CARETUP)		caretup
7 (CARETDOWN)	•	caretdown
8 (CARETLEFTBASE)	◀	caretleft (centered at base)
9 (CARETRIGHTBASE)	•	caretright (centered at base)
10 (CARETUPBASE)	_	caretup (centered at base)
11 (CARETDOWNBASE)	-	caretdown (centered at base)
"None", " " or ""		nothing
'\$\$'	f	Render the string using mathtext. E.g "\$f\$" for marker showing the letter f.
verts		A list of (x, y) pairs used for Path vertices. The center of the marker is located at (0, 0) and the size is normalized, such that the created path is encapsulated inside the unit cell.
path		A Path instance.
(numsides, 0, angle)		A regular polygon with numsides sides, rotated by angle.
(numsides, 1, angle)		A star-like symbol with numsides sides, rotated by angle.
(numsides, 2, angle)		An asterisk with numsides sides, rotated by angle.



#### Sử dụng linestyle:

https://matplotlib.org/stable/ gallery/lines\_bars\_and\_mar kers/linestyles.html

character	description
1.1	solid line style
' '	dashed line style
''	dash-dot line style
1:1	dotted line style

	Named linestyles		
solid 'solid'			
dotted 'dotted'			
dashed 'dashed'			
dashdot 'dashdot'			
	Parametrized linestyles		
loosely dotted (0, (1, 10))			
dotted (0, (1, 1))			
densely dotted			
loosely dashed			
dashed (0, (5, 5))			
densely dashed			
loosely dashdotted (0, (3, 10, 1, 10))			
dashdotted (0, (3, 5, 1, 5))			
densely dashdotted (0, (3, 1, 1, 1))			
dashdotdotted (0, (3, 5, 1, 5, 1, 5))			
loosely dashdotdotted 0, (3, 10, 1, 10, 1, 10))			
densely dashdotdotted (0, (3, 1, 1, 1, 1, 1))			



#### Kết hợp chỉ định color/marker/linestyle

```
fmt = '[color][marker][line]'
```

```
plt.plot([1, 2, 3], [1, 2, 3], 'go-')
plt.plot([4, 5, 6], [1, 2, 3], 'r^--')
```

```
plt.plot(x1, y1, 'g^', x2, y2, 'g-')
```

## 2.3 Gán nhãn cho thông số trong biểu đồ



Gán nhãn và bật/tắt chú giải (legend)

```
plt.plot('Nhãn-X', 'Nhãn-Y', data=[x,y])
plt.legend()
```

```
plt.plot([1, 2, 3], [1, 2, 3], 'go-', label='Loai 1')
plt.plot([4, 5, 6], [1, 2, 3], 'r^-', label='Loai 2')
plt.legend()
```

## 2.3 Gán nhãn cho thông số trong biểu đồ



Gán nhãn và bật/tắt chú giải (legend)

```
legend()
legend(labels)
legend(handles, labels)
```

```
Ví dụ:
ax.legend([line1, line2, line3], ['label1', 'label2', 'label3'])
```



## Phần 3. Các thao tác cơ bản khác

- 3.1. Figure size và tight\_layout
- 3.2. Các thao tác trên Axis
- 3.3. Hiển thị lưới của biểu đồ
- 3.4. Lưu file ảnh biểu đồ

## 3.1 Figure size và tight\_layout



- Figure size: figsize=(width, height in inches) (default: [6.4, 4.8])
- Tight\_layout: matplotlib tự căn chỉnh khoảng cách giữa các axes
  - plt.tight\_layout()
  - Hoặc thiết lập constrained layout=True

```
fig = plt.figure(figsize=(10,1))
plt.tight_layout()
```

## 3.2 Các thao tác trên Axis



❖ Có rất nhiều thao tác, dưới đây là các thao tác thường dùng:

Tầng Scripting	Tầng Artist	Mô tả
plt.axis(on/off)	<pre>ax.set_axis_on ax.set_axis_off ax.get_xaxis().set_v isible(False) ax2.get_yaxis().set_ visible(False)</pre>	Bật/tắt hiển thị trục tọa độ
<pre>plt.xlim plt.ylim</pre>	<pre>ax.set_xlim ax.set ylim</pre>	Thiết lập giới hạn của trục
<pre>plt.xlabel plt.ylabel</pre>	<pre>ax.set_xlabel ax.set ylabel</pre>	Thiết lập tiêu đề của trục tọa độ
plt.xticks	ax.set_xticks ax.set_yticks	Thiết lập giá trị trên trục tọa độ
plt.yticks	<pre>ax.set_xticklabels ax.set_yticklabels</pre>	Thiết lập nhãn hiển thị trên trục

## 3.2 Các thao tác trên Axis

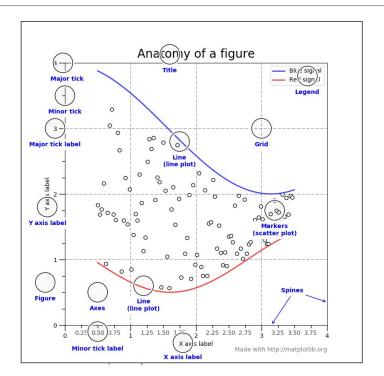


- Đảo ngược axis:
  - plt.gca().invert\_xaxis()/yaxis()
  - ax.invert\_xaxis()/yaxis()

## 3.2 Hiển thị lưới của biểu đồ



```
plt/ax.grid(linestyle="--", linewidth=0.5, color='.25')
```

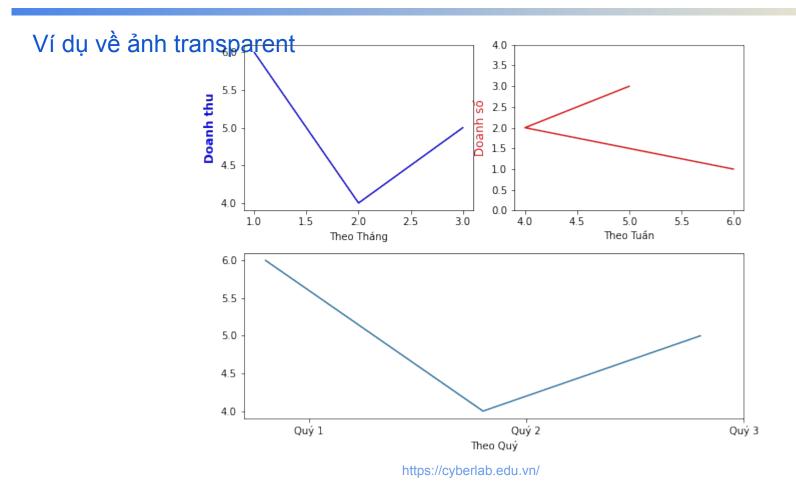


### 3.4 Lưu file ảnh của biểu đồ



Syntax: plt.savefig(tên-và-đường-dẫn-file-ảnh) Loại file ảnh (thông thường): png, jpg, svg





## Giải trí



#### David McCandless: Vẻ đẹp của hình ảnh hóa dữ liệu (18 phút)

https://www.ted.com/talks/david\_mccandless\_the\_beauty\_of\_data\_visualization?language=vi#t-1064702





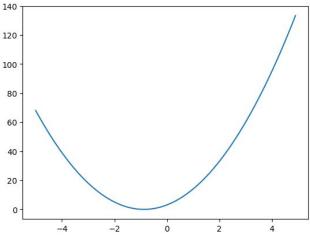
## **THANK YOU!**



## Demo và Bài tập



- Vẽ phương trình y = ax + b (cho phép gán giá trị khác nhau của a,b)
- 2. Đọc và Vẽ biểu đồ dữ liệu chứng khoán
- 3. Bài tập (nâng cao):
  - a. Vẽ biểu đồ dữ liệu chứng khoán với nhiều mã khác nhau
  - b. Tương tự bài 1, nhưng cho phương trình  $y=ax^2 + bx + c$  (-5 <= x <=5)



## Tham khảo



Nâng cao: links tham khảo tất cả các hàm API

- Pyplot: <a href="https://matplotlib.org/stable/api/\_as\_gen/matplotlib.pyplot.html#module-matplotlib.pyplot">https://matplotlib.org/stable/api/\_as\_gen/matplotlib.pyplot.html#module-matplotlib.pyplot</a>
- Figure: <a href="https://matplotlib.org/stable/api/figure\_api.html#matplotlib.figure.Figure">https://matplotlib.org/stable/api/figure\_api.html#matplotlib.figure.Figure</a>
- \* Axes: <a href="https://matplotlib.org/stable/api/axes">https://matplotlib.org/stable/api/axes</a> api.html#matplotlib.axes.Axes